Back-end Testing

Software

Postman

Postman is a popular API development tool designed to simplify and expedite the process of creating, testing, debugging, and sharing APIs. With an intuitive user interface, it allows developers, testers, and other stakeholders to effortlessly craft various types of HTTP requests, organize them into collections, and automate testing with powerful scripting capabilities. Postman facilitates collaboration by enabling users to share requests, collections, and environments, and it integrates seamlessly with a wide range of tools and services. Offering features such as environments, variables, and team libraries, Postman enhances productivity and ensures consistency across different testing environments. Additionally, it supports automation integration into CI/CD pipelines and basic performance testing functionalities, making it a versatile and indispensable tool for API development and testing workflows.

TestCase

Test Type	API Testing		Execution Type	**API Testing**
Objective	Verify users can login to account successfully			
Epic ID#	2	User story(Description)	Login	
Pre-condition				
Trail#	Test Case	Test Method	Test Data Input	Response(Result)
1		POST		

Test Type	API Testing		Execution Type	**API Testing**
Objective	Manage Program List			
Epic ID#	6	User story(Description)	The Program can be added, deleted and listed	
Pre-condition				
Trail#	Description	API	Method	Body	Response(Result)
1	get all programs	`http://localhost:8080/progs`	GET	None	{ "code": 1, "msg": "success", "data": [{ "progId": 1, "name": "program2", "maxCap": 10, "techReq":

					"Ipad",
					    "costPerson": 10.00,
					    "duration": "12:00-13:00",
					    "status": 1,
					    "description": "None",
					    "monday": 0,
					    "tuesday": 0,
					    "friday": 0,
					    "wednesday": 1,
					    "thursday": 0
					   }
					  ]
					}

2	search program by id	`http://localhost:8080/progs/1`	GET	None	{
					 "code": 1,
					 "msg": "success",
					 "data": [
					   {
					    "progId": 1,
					    "name": "program2",
					    "maxCap": 10,
					    "techReq": "Ipad",
					    "costPerson": 10.00,
					    "duration": "12:00-13:00",
					    "status": 1,
					    "description": "None",
					    "monday": 0,
					    "tuesday": 0,
					    "friday": 0,
					    "wednesday": 1,
					    "thursday": 0
					   }
					  ]
					}

3	delete a program	`http://localhost:8080/progs/1`	DELETE	None	{
					 "code": 1,
					 "msg": "success",
					 "data": null
					}

4	post a program	`http://localhost :8080/progs`	POST	Content-Type: application/json
				
				{
				  "name": "Test Program",
				  "maxCap": 10,
				  "techReq": "Some tech requirements",
				  "costPerson": 100,
				  "duration": "5",
				  "status": 1,
				  "Monday": 1,
				  "Tuesday": 0,
				  "Wednesday": 1,
				  "Thursday": 0,
				  "Friday": 1,
				  "description": "Some description"
				}
					{
					  "code": 1,
					  "msg": "success",
					  "data": null
					}

5	update a program	`http://localhost :8080/progs`	PUT	Content-Type: application/json
				
				{
				  "porg_id": 1,
				  "name": "Test Program2",
				  "maxCap": 10,
				  "techReq": "Some tech requirements",
				  "costPerson": 100,
				  "duration": "5",
				  "status": 1,
				  "Monday": 1,
				  "Tuesday": 0,
				  "Wednesday": 1,
				  "Thursday": 0,
				  "Friday": 1,
				  "description": "Some description"
				}
					{
					  "code": 1,
					  "msg": "success",
					  "data": null
					}